

WHAT IS CLAIMED IS:

- 1 1. A pipeline accelerator, comprising:
2 a memory; and
3 a hardwired-pipeline circuit coupled to the memory and operable to,
4 receive data,
5 load the data into the memory,
6 retrieve the data from the memory,
7 process the retrieved data, and
8 provide the processed data to an external source.
- 1 2. The pipeline accelerator of claim 1 wherein:
2 the memory is disposed on a first integrated circuit; and
3 the pipeline circuit is disposed on a second integrated circuit.
- 1 3. The pipeline accelerator of claim 1 wherein the pipeline circuit is disposed
2 on a field-programmable gate array.
- 1 4. The pipeline accelerator of claim 1 wherein the pipeline circuit is operable
2 to provide the processed data to the external source by:
3 loading the processed data into the memory;
4 retrieving the processed data from the memory; and
5 providing the retrieved processed data to the external source.
- 1 5. The pipeline accelerator of claim 1 wherein:
2 the external source comprises a processor; and
3 the pipeline circuit is operable to receive the data from the processor.
- 1 6. A computing machine, comprising:
2 a processor; and
3 a pipeline accelerator coupled to the processor and comprising,
4 a memory, and
5 a hardwired-pipeline circuit coupled to the memory and operable to,
6 receive data from the processor,
7 load the data into the memory,

8 retrieve the data from the memory,
9 process the retrieved data, and
10 provide the processed data to the processor.

1 7. A pipeline accelerator, comprising:
2 a memory; and
3 a hardwired-pipeline circuit coupled to the memory and operable to,
4 receive data,
5 process the received data,
6 load the processed data into the memory,
7 retrieve the processed data from the memory, and
8 provide the retrieved processed data to an external source.

1 8. A computing machine, comprising:
2 a processor; and
3 a pipeline accelerator coupled to the processor and comprising,
4 a memory, and
5 a hardwired-pipeline circuit coupled to the memory and operable to,
6 receive data from the processor,
7 process the received data,
8 load the processed data into the memory,
9 retrieve the processed data from the memory, and
10 provide the retrieved processed data to the processor.

1 9. A pipeline accelerator, comprising:
2 first and second memories; and
3 a hardwired-pipeline circuit coupled to the first and second memories and
4 comprising,
5 an input-data handler operable to receive raw data from an external
6 source and to load the raw data into the first memory,
7 a hardwired pipeline operable to process the raw data,

8 a pipeline interface operable to retrieve the raw data from the first memory,
9 provide the retrieved raw data to the hardwired pipeline, and load processed data
10 from the hardwired pipeline into the second memory, and

11 an output-data handler operable to retrieve the processed data from the
12 second memory and to provide the processed data to the external source.

1 10. The pipeline accelerator of claim 9 wherein:

2 the first and second memories each include respective first and second ports;

3 the input-data handler is operable to load the raw data via the first port of the first
4 memory,

5 the pipeline interface is operable to retrieve the raw data via the second port of
6 the first memory and to load the processed data via the first port of the second memory,
7 and

8 the output-data handler is operable to retrieve the processed data via the second
9 port of the second memory.

1 11. The pipeline accelerator of claim 9, further comprising:

2 a third memory coupled to the hardwired-pipeline circuit;

3 wherein the hardwired pipeline is operable to generate intermediate data while
4 processing the raw data; and

5 wherein the pipeline interface is operable to load the intermediate data into the
6 third memory and to retrieve the intermediate data from the third memory.

1 12. The pipeline accelerator of claim 9 wherein:

2 the first and second memories are respectively disposed on first and second
3 integrated circuits; and

4 the pipeline circuit is disposed on a field-programmable gate array.

1 13. The pipeline accelerator of claim 9, further comprising:

2 an input-data queue coupled to the input-data handler and the pipeline interface;

3 wherein the input-data handler is operable to load into the input-data queue a
4 pointer to a location of the raw data within the first memory; and

5 wherein the pipeline interface is operable to retrieve the raw data from the
6 location using the pointer.

1 14. The pipeline accelerator of claim 9, further comprising:
2 an output-data queue coupled to the output-data handler and the pipeline
3 interface;

4 wherein the pipeline interface is operable to load into the output-data queue a
5 pointer to a location of the processed data within the second memory; and

6 wherein the output-data handler is operable to retrieve the processed data from
7 the location using the pointer.

1 15. The pipeline accelerator of claim 9, further comprising:

2 wherein each of the input-data handler, hardwired pipeline, pipeline interface,
3 and output-data handler has a respective operating configuration; and

4 a configuration manager coupled to and operable to set the operating
5 configurations of the input-data handler, hardwired pipeline, pipeline interface, and
6 output-data handler.

1 16. The pipeline accelerator of claim 9, further comprising:

2 wherein each of the input-data handler, hardwired pipeline, pipeline interface,
3 and output-data handler has a respective operating status; and

4 an exception manager coupled to and operable to identify an exception in the
5 input-data handler, hardwired pipeline, pipeline interface, or output-data handler in
6 response to the operating statuses.

1 17. A pipeline accelerator, comprising:

2 a hardwired pipeline operable to process data; and

3 an input-data handler coupled to the hardwired pipeline and operable to,

4 receive the data,

5 determine whether the data is directed to the hardwired pipeline, and

6 provide the data to the hardwired pipeline if the data is directed to the

7 hardwired pipeline.

1 18. The pipeline accelerator of claim 17 wherein the input-data handler is
2 further operable to:
3 receive the data by,
4 receiving a message that includes a header and the data, and
5 extracting the data from the message; and
6 determine whether the data is directed to the hardwired pipeline by analyzing the
7 header.

1 19. The pipeline accelerator of claim 17 wherein the hardwired pipeline and
2 the input-data handler are disposed on a single field-programmable gate array.

1 20. The pipeline accelerator of claim 17 wherein the hardwired pipeline and
2 the input-data handler are disposed on respective field-programmable gate arrays.

1 21. A computing machine, comprising:
2 a processor; and
3 a pipeline accelerator coupled to the processor and comprising,
4 a hardwired pipeline operable to process data, and
5 an input-data handler coupled to the hardwired pipeline and operable to,
6 receive the data from the processor,
7 determine whether the data is directed to the hardwired pipeline,
8 and
9 provide the data to the hardwired pipeline if the data is directed to
10 the hardwired pipeline.

1 22. A pipeline accelerator, comprising:
2 a hardwired pipeline operable to generate data; and
3 an output-data handler coupled to the hardwired pipeline and operable to,
4 receive the data,
5 determine a destination of the data, and
6 provide the data to the destination.

1 23. The pipeline accelerator of claim 22 wherein the output-data handler is
2 further operable to:

3 determine the destination of the data by,
4 identifying a type of the data, and
5 determining the destination based on the type of the data; and
6 provide the data to the destination by,
7 generating a message that identifies the destination and that includes the
8 data, and
9 providing the message to the destination.

1 24. A computing machine, comprising:
2 a processor operable to execute threads of an application; and
3 a pipeline accelerator coupled to the processor and comprising:
4 a hardwired pipeline operable to generate data, and
5 an output-data handler coupled to the hardwired pipeline and operable to,
6 receive the data,
7 identify a thread of the application that subscribes to the data, and
8 provide the data to the subscribing thread.

1 25. A pipeline accelerator, comprising:
2 a hardwired pipeline operable to process data values; and
3 a sequence manager coupled to and operable to control the operation of the
4 hardwired pipeline.

1 26. The pipeline accelerator of claim 25 wherein the sequence manager is
2 operable to control an order in which the hardwired pipeline receives the data values.

1 27. The pipeline accelerator of claim 25 wherein the sequence manager is
2 further operable to:
3 receive an event; and
4 control the hardwired pipeline in response to the event.

1 28. The pipeline accelerator of claim 25 wherein the sequence manager is
2 further operable to:
3 receive a synchronization signal; and

4 control the operation of the hardwired pipeline in response to the synchronization
5 signal.

1 29. The pipeline accelerator of claim 25 wherein the sequence manager is
2 further operable to:

3 sense an occurrence relative to the hardwired pipeline; and
4 generate an event in response to the occurrence.

1 30. A computing machine, comprising:

2 a processor operable to generate data and an event; and

3 a pipeline accelerator coupled to the processor and comprising,

4 a hardwired pipeline operable to receive the data from the processor and
5 process the received data; and

6 a sequence manager coupled to the hardwired pipeline and operable to
7 receive the event from the processor and to control the operation of the
8 hardwired pipeline in response to the event.

1 31. A pipeline accelerator, comprising:

2 a hardwired-pipeline circuit having an operating configuration and operable to
3 process data; and

4 a configuration manager coupled to the hardwired-pipeline circuit and operable to
5 set the operating configuration.

1 32. The pipeline accelerator of claim 31 wherein:

2 the hardwired-pipeline circuit includes a configuration register; and

3 the configuration manager is operable to set the operating configuration by
4 loading a configuration value into the configuration register.

1 33. The pipeline accelerator of claim 32 wherein the configuration manager is
2 operable to receive the configuration value from an external source.

1 34. A computing machine, comprising:

2 a processor operable to generate data and a configuration value; and

3 pipeline accelerator coupled to the processor and comprising,

4 a hardwired-pipeline circuit having an operating configuration and operable
5 to process the data, and

6 a configuration manager coupled to the hardwired-pipeline circuit and
7 operable to set the operating configuration in response to the configuration value.

1 35. A pipeline accelerator, comprising:

2 a hardwired-pipeline circuit having an operating status and operable to process
3 data; and

4 an exception manager coupled to the hardwired-pipeline circuit and operable to
5 identify an exception in the operation status of the hardwired-pipeline circuit in response
6 to the operating status.

1 36. The pipeline accelerator of claim 35 wherein:

2 the hardwired-pipeline circuit is operable to generate a status value that
3 represents the operating status; and

4 the exception manager is operable to identify the exception in response to the
5 status value.

1 37. The pipeline accelerator of claim 36 wherein:

2 the hardwired-pipeline circuit includes a status register that is operable to store
3 the status value; and

4 the exception manager receives the status value from the status register.

1 38. The pipeline accelerator of claim 35 wherein the exception manager is
2 operable to identify an exception in the operating status of the hardwired-pipeline circuit
3 to an external source.

1 39. A computing machine, comprising:

2 a processor operable to generate data; and

3 a pipeline accelerator, comprising,

4 a hardwired-pipeline circuit having an operating status and operable to
5 process data and to generate a status value that represents the operating status,
6 and

7 an exception manager coupled to the hardwired-pipeline circuit and
8 operable to identify an exception in the operating status of the hardwired-pipeline
9 circuit in response to the status value and to notify the processor of the
10 exception.

1 40. A computing machine, comprising:
2 a pipeline accelerator, comprising,
3 a hardwired-pipeline circuit having an operating status and operable to
4 process data, and
5 an exception manager coupled to the hardwired-pipeline circuit and
6 operable to generate a status value that represents the operating status; and
7 a processor coupled to the pipeline accelerator and operable to generate the
8 data, to receive the status value, and to determine whether the hardwired-pipeline
9 circuit is malfunctioning by analyzing the status value.

1 41. A method, comprising:
2 loading data into a memory;
3 retrieving the data from the memory;
4 processing the retrieved data with a hardwired-pipeline circuit; and
5 providing the processed data to an external source.

1 42. The method of claim 41 wherein providing the processed data comprises:
2 loading the processed data into the memory;
3 retrieving the processed data from the memory; and
4 providing the retrieved processed data to the external source.

1 43. A method, comprising:
2 processing data with a hardwired-pipeline circuit;
3 loading the processed data into a memory;
4 retrieving the processed data from the memory; and
5 providing the retrieved processed data to an external source.

1 44. A method, comprising:
2 loading raw data from an external source into a first memory;

3 retrieving the raw data from the first memory;
4 processing the retrieved data with a hardwired pipeline;
5 loading the processed data from the hardwired pipeline into a second memory;
6 and
7 providing the processed data from the second memory to the external source.

1 45. The method of claim 44 wherein:

2 loading the raw data comprises loading the raw data via a first port of the first
3 memory;
4 retrieving the raw data comprises retrieving the raw data via a second port of the
5 first memory;
6 loading the processed data comprises loading the processed data via a first port
7 of the second memory; and
8 providing the processed data comprises retrieving the processed data via a
9 second port of the second memory.

1 46. The method of claim 44, further comprising:

2 generating intermediate data with the hardwired pipeline in response to
3 processing the raw data;
4 loading the intermediate data into a third memory; and
5 providing the intermediate data from the third memory back to the hardwired
6 pipeline.

1 47. The method of claim 44, further comprising:

2 loading into an input-message queue a pointer to a location of the raw data within
3 the first memory; and
4 wherein retrieving the raw data comprises retrieving the raw data from the
5 location using the pointer.

1 48. The method of claim 44, further comprising:

2 loading into an output-message queue a pointer to a location of the processed
3 data within the second memory; and

4 wherein retrieving the processed data comprises retrieving the processed data
5 from the location using the pointer.

1 49. The method of claim 44, further comprising setting parameters for loading
2 and retrieving the raw data, processing the retrieved data, and loading and providing the
3 processed data.

1 50. The method of claim 44, further comprising determining whether an error
2 occurs during the loading and retrieving of the raw data, the processing of the retrieved
3 data, and the loading and providing of the processed data.

1 51. A method, comprising:
2 receiving data;
3 determining whether the data is directed to a hardwired pipeline; and
4 providing the data to the hardwired pipeline if the data is directed to the
5 hardwired pipeline.

1 52. The method of claim 51 wherein:
2 receiving the data comprises,
3 receiving a message that includes a header and the data, and
4 extracting the data from the message; and
5 determining whether the data is directed to the hardwired pipeline comprises
6 analyzing the header.

1 53. A method, comprising:
2 generating data with a hardwired pipeline;
3 determining a destination of the data; and
4 providing the data to the destination.

1 54. The method of claim 53 wherein:
2 determining the destination of the data comprises,
3 identifying a type of the data, and
4 determining the destination based on the type of the data; and
5 providing the data to the destination comprises,

6 generating a message that identifies the destination and that includes the
7 data, and
8 providing the message to the destination.

1 55. A method, comprising:
2 processing data values with a hardwired pipeline; and
3 sequencing the operation of the hardwired pipeline.

1 56. The method of claim 55 wherein sequencing the operation comprises
2 sequencing an order in which the hardwired pipeline processes the data values.

1 57. The method of claim 55 wherein sequencing the operating comprises
2 synchronizing the operation of the hardwired pipeline to a synchronization signal.

1 58. The method of claim 55, further comprising:
2 sensing a predefined occurrence during operation of the hardwired pipeline; and
3 generating an event in response to the occurrence.

1 59. A method, comprising:
2 loading a configuration value into a register; and
3 setting an operating configuration of a hardwired pipeline with the configuration
4 value.

1 60. A method, comprising:
2 processing data with a hardwired pipeline; and
3 identifying an error in the processed data by analyzing an operating status of the
4 hardwired pipeline.

1 61. A method for designing a hardwired-pipeline circuit, comprising:
2 retrieving from a library a first data representation of a communication interface;
3 generating a second data representation of a hardwired pipeline that is to be
4 coupled to the communication interface; and
5 combining the first and second data representations to generate
6 hard-configuration data for the hardwired-pipeline circuit.

1 62. The method of claim 61, further comprising modifying the first data
2 representation by selecting values for predetermined parameters of the services layer
3 before combining the first and second data representations.

1 63. The method of claim 61 wherein the communication interface is operable
2 to allow the hardwired-pipeline circuit to communicate with another circuit.

1 64. The method of claim 61 wherein combining the first and second data
2 representations comprises compiling the first and second data representations into the
3 hard-configuration data.

1 65. The method of claim 61 wherein the hard-configuration data comprises
2 firmware.